

TAMALPAIS UNION HIGH SCHOOL DISTRICT
Larkspur, California

Course of Study

COMPUTER PROGRAMMING 1-6

I. INTRODUCTION

Computer Programming 1-6 is a sequence of one semester elective courses. The format allows students who wish to study one particular programming language in greater depth to do so but also allows students to explore different languages each semester.

These courses are designed for students with a strong interest in Programming and its applications. Good logic skills and perseverance are stressed. Depending on availability, student interest and teacher background, languages that may be studied include: Visual Basic, C++, Visual C++, Java, C#, PHP, Open GL, Flash ActionScript, SQL.

The first goal of this course is to develop students' understanding of the basic concepts of programming, regardless of language. The second goal is to apply these concepts in a particular programming language.

This course complements advanced web design and computer graphics coursework as a great deal of web design includes programming skills and concepts. Flash animation in Computer Graphics now involves extensive use of Flash ActionScript programming.

This course addresses the following Tam 21st Century goals:

- Acquire, manage and use knowledge and skills
- Think critically and creatively
- Practice decision making and problem solving
- Encourage student creativity
- Students should have opportunities to pursue their individual goals and aspirations
- Flexibility and choice enhance learning
- Acquisition of technological skills is essential to success

This course addresses the following Student Learning Outcomes:

1. Communicate articulately, effectively, and persuasively when speaking and writing.
2. Read and analyze material in a variety of disciplines.
3. Use technology as a tool to access information, analyze and solve problems, and communicate ideas.
4. Apply mathematical knowledge and skills to analyze and solve problems.
5. Demonstrate school-to-work/post-secondary transition skills and knowledge.

This course expands students' technological proficiency and prepares them for collegiate level computer science studies and/or entry level work in the computer industry.

II. STUDENT LEARNING OUTCOMES

A. Upon completion of this course, students will (regardless of the specific programming language):

1. Read and interpret programming code.
2. Write and format logical, clear code.
3. Create user-friendly code.
4. Create programmer-friendly code.
5. Troubleshoot and debug programs.
6. Use arrays, loops, conditional statements, decision statements, recursive functions.
7. Create programs that save information to disk and access information from disk.
8. Update and alter code.
9. Create modular programs.
10. Use at least one sorting algorithm.
11. Correctly use string manipulation.
12. Correctly use data types.
13. Effectively work with other students.
14. Describe the benefits and drawbacks of a career in Computer Programming.

At the end of Programming 2, all students complete an assignment showing proficiency in all of the above outcomes.

Programming 3 - 6 is more of an independent study approach to continued programming. Students have the opportunity to achieve the above outcomes in a different language than in Programming 1-2. They may also choose to apply the outcomes they have mastered in Programming 1-2 to more complex programming tasks, many of which have real-world applications.

Examples of student projects include real-time game programs and simulations, robotics, business accounting programs (spreadsheet and database types), gradebook programs, Triangle angle solver, Text-based adventure game, Creating a 3-D world, Asteroids-like live-action game, Arithmetic expression parser/evaluator, DOS simulator, Checkers, Word search puzzle creator/solver, Space invaders, Network chess, Duck hunt, Lots of arcade-type games, and online web-based applets. In Programming 3-6, students complete at least one culminating assignment that simulates a real-world, workplace project.

B. Students will cover the following state subject Content Standards:

Information Technology Industry Sector
3.0 Career Planning and Management

- 3.1 Know the personal qualifications, interests, aptitudes, knowledge, and skills necessary to succeed in careers.
 - 3.5 Understand the past, present, and future trends that affect careers, such as technological developments and societal trends, and the resulting need for lifelong learning.
 - 3.7 Explore career opportunities in business through such programs as virtual enterprise, work experience, and internships.
- 4.0 Technology
- 4.1 Understand past, present, and future technological advances as they relate to a chosen pathway.
 - 4.2 Understand the use of technological resources to gain access to, manipulate, and produce information, products, and services.
 - 4.3 Understand the influence of current and emerging technology on selected segments of the economy.
- 5.0 Problem Solving and Critical Thinking
- 5.3 Use critical thinking skills to make informed decisions and solve problems.
- 9.0 Leadership and Teamwork
- 9.1 Understand the characteristics and benefits of teamwork, leadership, and citizenship in the school, community, and workplace settings.
 - 9.3 Understand how to organize and structure work individually and in teams for effective performance and the attainment of goals.
- 10.0 Technical Knowledge and Skills
- 10.1 Know how to use a variety of business- and industry-standard software and hardware, including major proprietary and open standards.
 - 10.3 Understand the economic effects of technology on a business in the global marketplace.
 - 10.7 Analyze the functions, features, and limitations of different operating systems, environments, applications, and utilities.
- 11.0 Demonstration and Application
- A. Information Support and Services Pathway
 - A7.0 Students understand software applications and life-cycle phases:
 - A7.1 Know common industry-standard software and its applications.
 - A7.3 Know a variety of sources for reference materials (e.g., online help, vendors' Web sites, online discussion groups, tutorials, manuals).
 - A7.5 Know current and emerging industry-standard

technology and trends.

A8.0 Students understand the importance of reading, writing, and comprehending documentation in a technical environment:

A8.1 Know appropriate search procedures for different types of information, sources, and queries.

A8.2 Evaluate the accuracy, relevance, and comprehensiveness of retrieved information.

A8.3 Analyze the effectiveness of online information resources to support collaborative tasks, research, publications, communications, and increased productivity.

B. Media Support and Services Pathway

B5.10 Differentiate among various versions of Internet programming languages.

D. Programming and Systems Development Pathway

D1.0 Students understand the strategies necessary to define and analyze systems and software requirements:

D1.1 Develop information technology-based strategies and project plans to solve specific problems.

D1.2 Know how systems and software requirements are determined in various situations.

D1.3 Know the effective use of tools for software development.

D1.4 Know the software development process.

D2.0 Students understand programming languages:

D2.1 Know the fundamentals of programming languages and concepts.

D2.2 Compare programs by using control structures, procedures, functions, parameters, variables, error recovery, and recursion.

D3.0 Students understand the creation and design of a software program:

D3.1 Analyze customers' needs and requirements for software.

D3.2 Know how specifications and codes are developed for new and existing software applications.

D3.4 Know multiple ways in which to store, retrieve, and access information.

D4.0 Students understand the process of testing, debugging, and maintaining programs to meet specifications:

D4.1 Know the steps involved in the software-testing process.

- D4.3 Know how different systems and associated utilities perform such functions as file management, backup and recovery, and execution of programs.
- D5.0 Students understand the importance of quality assurance tasks in producing effective and efficient products:
 - D5.1 Know the standards and requirements for software quality assurance.
 - D5.2 Know common quality assurance tasks and their place in the development process.
 - D5.3 Understand the ways in which specification changes and technological advances can require the modification of programs.
 - D5.4 Know various sorting and searching methods and their comparative advantages.
 - D5.5 Know the characteristics of reliable, effective, and efficient products.
- D6.0 Students understand the importance of effective interfaces in the interaction between humans and computer systems:
 - D6.1 Understand how to support access, privacy, and high ethical standards in computing.
 - D6.2 Use knowledge of cognitive, physical, and social interactions to create and design user

III. UNITS OF INSTRUCTION – UbD work not completed yet

IV. METHODS/MATERIALS

A. Methods

This course uses a combination of lectures, demonstrations, text readings and assignments. A significant amount of time is “hands-on” doing work on the computer. At least two-thirds of the time will be spent on the computer.

B. Materials

Students will use current software, textbooks and tutorials (both in print and on-line). Textbooks are selected from the Board Approved Textbook list. These materials include:

- Introduction to Computer Science Using C++; Todd Knowlton; South-Western Publishing; 1998
- David Goldsmith’s Redwood Programming Web Site (<http://rhsweb.org/assignments/Goldsmith/>)
- C++ Resources Network (<http://www.cplusplus.com/>)

- Visual Basic Developers Resource Network
(<http://www.mvps.org/vbnet/>)
- Java Development Resources (<http://www.developer.com/java/>)
- OpenGL Resources (<http://www.opengl.org/>)

C. Technology

This course is conducted using up-to-date computers and software.

D. School-to-Career

Guest speakers, job shadows, adult mentors and/or real-world projects will be used to increase student awareness of the connection between this course and the world of work.

E. Suggested Instructional Time Allocation

How Computers Are Programmed	1 week
Entering, Compiling and Running a Program	1 week
Variables and Constants	2 weeks
Math Operators	2 weeks
Strings and Screen I/O	2 weeks
Decision-Making	2 weeks
Loops	2 weeks
Functions	2 weeks
Pointers/Structures	2 weeks
Data Files	2 weeks
Object-Oriented Programming	2 weeks
String Functions	2 weeks
Arrays, Templates, Vectors	2 weeks
Multi-Dimensional Arrays and Matrices	2 weeks
Sorting	2 weeks
Linked Lists	2 weeks
Queues, Stacks and Trees	2 weeks
Recursion and Searching	2 weeks

V. ANCHORS OF STUDENT WORK

For the purposes of evaluation and grading during the course, homework assignments, in-class exercises, projects, quizzes and exams will be used. Students will be informed of the grading criteria in writing at the beginning of each semester.

VI. TROUBLE-SHOOTING GUIDE

This course requires proper understanding and setup of the necessary software. Most commonly used software has included:

Microsoft Visual Studio 6
Microsoft .NET
Alice (www.alice.org).

VII. COURSE ASSESSMENT

Computer Programming teachers will meet annually to discuss successes and difficulties in an effort to improve the effectiveness of the course. Student performance data, as it becomes necessary, will be analyzed by teachers. Enrollment and retention data will be collected to assess needed adjustments. Teachers will meet to share strategies and discuss modifications.

VIII. GENERAL INFORMATION

A. Prerequisites

There are no prerequisites for Computer Programming 1. To take Computer Programming 2-6, students need to complete the previous course(s) in the sequence or receive permission from the course instructor. A strong background or enjoyment in math is not required but may be helpful.

B. Requirements Met

This course may be used as elective credit towards graduation but does not meet any specific graduation requirement.

This course is accepted towards the “G” requirement for UC admissions. It is also accepted for the CSU “G” requirement.

Approved:
Revised: 6/98, 6/03, 10/03, 5/07, 6/09